

## **Lección 03:**

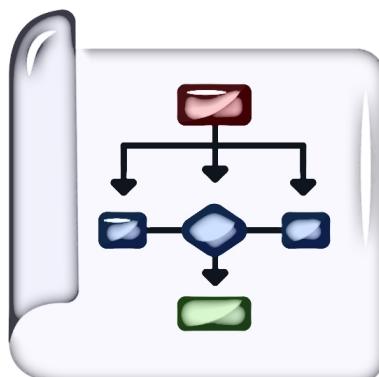
### **Estructuras de control condicionales**

Fredy Camacho García  
Jose Alonso Oviedo  
Andrés Mauricio Arciniegas

# 3

## Estructuras de control condicionales

Las estructuras lógicas condicionales o selectivas se encuentran en la solución algorítmica de casi todo tipo de problemas. Se emplean cuando en el desarrollo de la solución de un problema se debe tomar una decisión con el fin de establecer un proceso o señalar un camino alternativo a seguir, es decir, si ejecuta o no un grupo de instrucciones.



El algoritmo se sigue ejecutando de arriba hacia abajo, como ocurre con las **estructuras de control secuenciales**, sin embargo, puede ocurrir que durante la ejecución algunas instrucciones no se ejecuten, dependiendo de una **condición**.

Una **condición** se refiere a una comparación que se establece entre los datos con los cuales opera el algoritmo.

Para establecer una comparación, se emplean los **operadores relacionales** (o de comparación). A diferencia de los operadores aritméticos, el resultado que se obtiene al utilizarlos en una expresión es un valor lógico: Verdadero (True) o Falso (False). Los valores lógicos se conocen también como **booleanos**.

Los operadores relacionales en Python son los siguientes:

Operador	Significado
<	Menor que
>	Mayor que
<=	Menor o igual que
>=	Mayor o igual que
==	Igual que
!=	Diferente (Distinto) de

A las comparaciones se les conoce como **Expresiones Lógicas**.

## Diferencia entre los operadores de comparación y asignación

El **operador de asignación** es aquel que se representa mediante un solo igual. Por ejemplo, en la instrucción,

`a = 5`

se establece el valor de la variable **a** en **5**, esto es, el valor de la variable **a** cambió

El **operador de comparación** es aquel que se representa mediante el doble igual. Por ejemplo, en la instrucción,

`a == 5`

se compara el valor de la variable **a** con **5**, esto es, el valor de la variable **a** no cambia y en cambio se pregunta si **a** es igual a **5**.

## Tipos de estructuras condicionales



En términos generales, se habla de cuatro tipos de estructuras condicionales:

- Simples
- Dobles
- Múltiples en cascada
- Por casos (No soportada por Python)

A continuación, se detallará cada una de estas con ejemplos desarrollados en Python mediante el entorno de desarrollo en línea **Google Colaboratory**.

## Estructuras condicionales simples



Es aquella instrucción en la cual se establece que se va a realizar si la comparación (condición) es **verdadera (True)** únicamente.

Las instrucciones condicionales simples tienen la siguiente estructura, a nivel de algoritmos:

**Si** condición **Entonces**

    Instrucciones por ejecutar si condición es **True**

**Fin Si**

En Python, la instrucción condicional:

```
if condición :
```

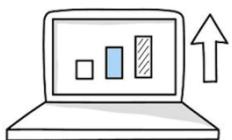
```
    instrucción 1 por ejecutar si condición es True
```

```
    instrucción 2 por ejecutar si condición es True
```

```
    instrucción 3 por ejecutar si condición es True
```

```
    . . .
```

```
otra instrucción fuera de if
```



## Ejemplo 1.

Simular un semáforo desplegando un mensaje en pantalla que indica que puede pasar si la luz está en color verde.

### Solución:

```
# Ejemplo 1 - Condicionales

colorLuz = 'verde'

if colorLuz == 'verde':
    print( 'Puede pasar' )

print( 'Buen viaje' )
```

Observe que la instrucción `if colorLuz == 'verde'`: establece una comparación entre la variable `colorLuz` y el valor `'verde'`. Si esta comparación da como resultado **True** (Verdadero), es decir, son iguales, entonces se ejecutan las instrucciones que están debajo con **indentación** (espacios a la izquierda, o, sangría).

En este caso la instrucción `print( 'Puede pasar' )` se ejecuta y, por tanto, se muestra el mensaje en pantalla.

Una vez aparezca una instrucción sin **indentación**, significa que el bloque condicionado finalizó. En este caso, la instrucción `print( 'Buen viaje' )` está al mismo nivel (justificado a la izquierda) de la instrucción `if`, por tanto, la finaliza.

La implementación en Python con su respectiva ejecución se observa a continuación:

```

# Ejemplo 1 - Condicionales

colorLuz = 'verde'

if colorLuz == 'verde':
    print( 'Puede pasar' )

print( 'Buen viaje' )

```

→ Puede pasar  
Buen viaje

En el siguiente ejercicio, observe que no se muestra el mensaje "Puede pasar" porque la comparación da como resultado un valor **False**.

```

colorLuz = 'rojo'

if colorLuz == 'verde':
    print( 'Puede pasar' )

print( 'Buen viaje' )

```

La implementación en Python y posterior ejecución despliega el siguiente resultado:

```

[6] colorLuz = 'rojo'

if colorLuz == 'verde':
    print( 'Puede pasar' )

print( 'Buen viaje' )

```

Buen viaje



## Indentación

En Python, la **indentación** es fundamental. Observe la ejecución del siguiente código:

```

    colorLuz = 'rojo'

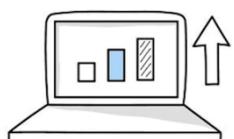
    if colorLuz == 'verde':
        print( 'Puede pasar' )

        print( 'Buen viaje' )

```

Las dos instrucciones `print` están al mismo nivel (es decir, tienen la misma indentación), y, por tanto, están condicionadas ambas. Por tanto, se ejecutarán si la condición es verdadera. En este caso, no se mostrará mensaje alguno en pantalla.

### Ejemplo 2.



Escribir un script que lea la edad de una persona y determine si puede ingresar a un bar. Para ingresar a un bar debe ser mayor de edad. La edad debe ser leída por teclado.

### Solución

- *Datos Entrada:* edad
- *Datos Salida:* mensaje

```

# Ejemplo 2 - Condicionales simples

print( "Ingrese su edad:" )
edad = int( input() )

if edad >= 18 :
    print( "Puede ingresar al bar" )

```

### Caso de Prueba 1: Comparación con resultado True

```
# Ejemplo 2 - Condicionales simples

print( "Ingrese su edad: " )
edad = int( input() )

if edad >= 18 :
    print( "Puede ingresar al bar" )

print( "Fin del algoritmo" )

→ Ingrese su edad:
20
Puede ingresar al bar
Fin del algoritmo
```

### Caso de Prueba 2: Comparación con resultado False

```
# Ejemplo 2 - Condicionales simples

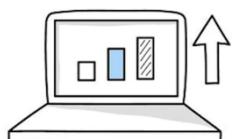
print( "Ingrese su edad: " )
edad = int( input() )

if edad >= 18 :
    print( "Puede ingresar al bar" )

print( "Fin del algoritmo" )

Ingrese su edad:
17
Fin del algoritmo
```

### Ejemplo 3.



Escribir un script que lea la calificación de un estudiante en un examen y determine si lo aprobó. Para tal fin, un examen se aprueba si su calificación es mayor a 5. La calificación está en el rango de 1 a 10.

### Solución

- Datos Entrada: calificacion
- Datos Salida: mensaje

```
# Ejemplo 3 - Condicionales simples

print( "Ingrese le calificación: " )
calificacion = int( input() )

if calificacion > 5 :
    print( "Aprobó el examen" )
```

*Caso de prueba 1: Comparación con resultado True*



```
# Ejemplo 3 - Condicionales simples

print( "Ingrese le calificación: " )
calificacion = int( input() )

if calificacion > 5 :
    print( "Aprobó el examen" )

print( "Fin del algoritmo" )

Ingrese le calificación:
8
Aprobó el examen
Fin del algoritmo
```

*Caso de prueba 2: Comparación con resultado False*



```
# Ejemplo 3 - Condicionales simples

print( "Ingrese le calificación: " )
calificacion = int( input() )

if calificacion > 5 :
    print( "Aprobó el examen" )

print( "Fin del algoritmo" )

Ingrese le calificación:
4
Fin del algoritmo
```

*Caso de Prueba 3: Comparación con resultado False donde los dos valores son iguales*

```
# Ejemplo 3 - Condicionales simples

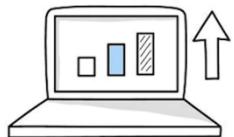
print( "Ingrese le calificación: " )
calificación = int( input() )

if calificación > 5 :
    print( "Aprobó el examen" )

print( "Fin del algoritmo" )

Ingrese le calificación:
5
Fin del algoritmo
```

### Ejemplo 4.



Escriba un script en el cual, dado como dato el sueldo de un trabajador, aplique un aumento del 5% si su sueldo es inferior a \$1.000.000. Imprima en este caso el nuevo sueldo del trabajador.

Por ejemplo:

Si el sueldo es 800000, entonces el nuevo sueldo es:  
 $800000 * 5 / 100 = 40000$  Esto representa el incremento  
 $800000 + 40000 = 840000$  Esto es el nuevo sueldo

### Solución

- *Datos Entrada:*      sueldoActual
- *Datos Salida:*      nuevoSueldo

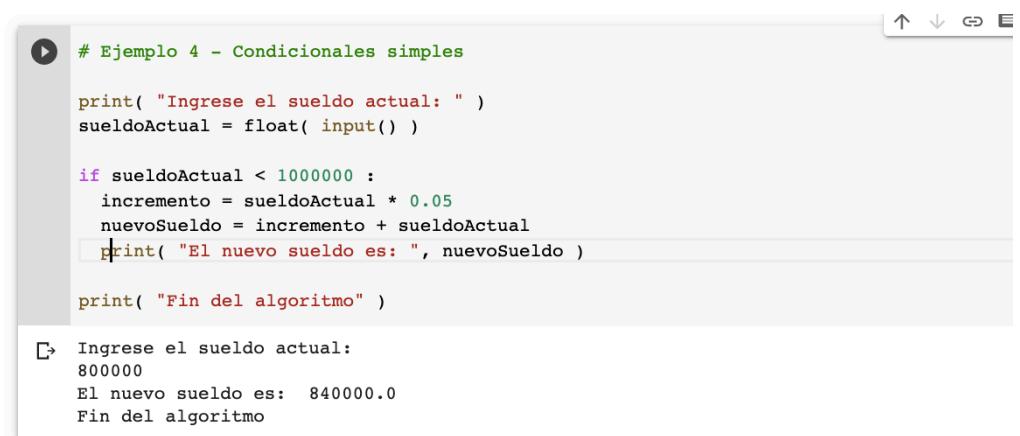
El código en Python es el siguiente:

```
# Ejemplo 4 - Condicionales simples

print( "Ingrese el sueldo actual: " )
sueldoActual = float( input() )

if sueldoActual < 1000000 :
    incremento = sueldoActual * 0.05
    nuevoSueldo = incremento + sueldoActual
    print( "El nuevo sueldo es: ", nuevoSueldo )
```

### Caso de Prueba 1: Comparación con resultado True



```
# Ejemplo 4 - Condicionales simples

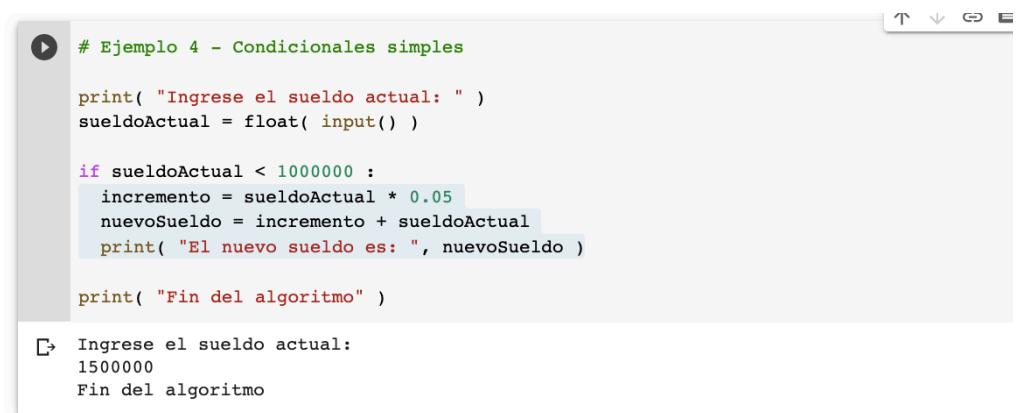
print( "Ingrese el sueldo actual: " )
sueldoActual = float( input() )

if sueldoActual < 1000000 :
    incremento = sueldoActual * 0.05
    nuevoSueldo = incremento + sueldoActual
    print( "El nuevo sueldo es: ", nuevoSueldo )

print( "Fin del algoritmo" )

Ingrese el sueldo actual:
800000
El nuevo sueldo es: 840000.0
Fin del algoritmo
```

### Caso de Prueba 2: Comparación con resultado False



```
# Ejemplo 4 - Condicionales simples

print( "Ingrese el sueldo actual: " )
sueldoActual = float( input() )

if sueldoActual < 1000000 :
    incremento = sueldoActual * 0.05
    nuevoSueldo = incremento + sueldoActual
    print( "El nuevo sueldo es: ", nuevoSueldo )

print( "Fin del algoritmo" )

Ingrese el sueldo actual:
1500000
Fin del algoritmo
```

## Estructuras condicionales dobles



Es aquella instrucción en donde se establece que se va a realizar si la comparación (condición) es **verdadera (True)**, y, se establece también que se va a realizar si es **falsa (False)**

Las instrucciones condicionales dobles tienen la siguiente estructura, a nivel de algoritmos:

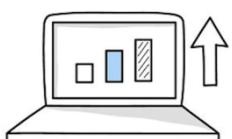
```
Si condición Entonces
    Instrucciones por ejecutar si condición es True
Sino
    Instrucciones por ejecutar si condición es False
Fin Si
```

En Python, se establece de la siguiente forma:

```
if condición :
    instrucciones a ejecutar si condición es True
else :
    instrucciones a ejecutar si condición es False
```

Solo se ejecuta uno de los dos bloques: o el bloque **if** o el bloque **else**, pero jamás los dos.

### Ejemplo 5.



Modifique el ejemplo 1 para que considere mostrar un mensaje de "Alto" si la luz del semáforo está en rojo.

### Solución

- *Datos Entrada:* colorLuz
- *Datos Salida:* mensaje

```
# Ejemplo 5 - Condicionales dobles

colorLuz = 'rojo'

if colorLuz == 'verde':
    print( 'Puede pasar' )
else:
    print( "Alto" )
```

*Caso de Prueba 1: Comparación con resultado False*



```
# Ejemplo 5 - Condicionales dobles

colorLuz = 'rojo'

if colorLuz == 'verde':
    print( 'Puede pasar' )
else:
    print( "Alto" )

print( 'Buen viaje' )
```

Alto  
Buen viaje

El caso de prueba cuando la comparación es True fue expuesto en el ejemplo 1.

### Ejemplo 6.

Modifique el ejercicio 2 para que ahora también despliegue un mensaje de que no es posible el ingreso a un bar si es menor de edad

### Solución

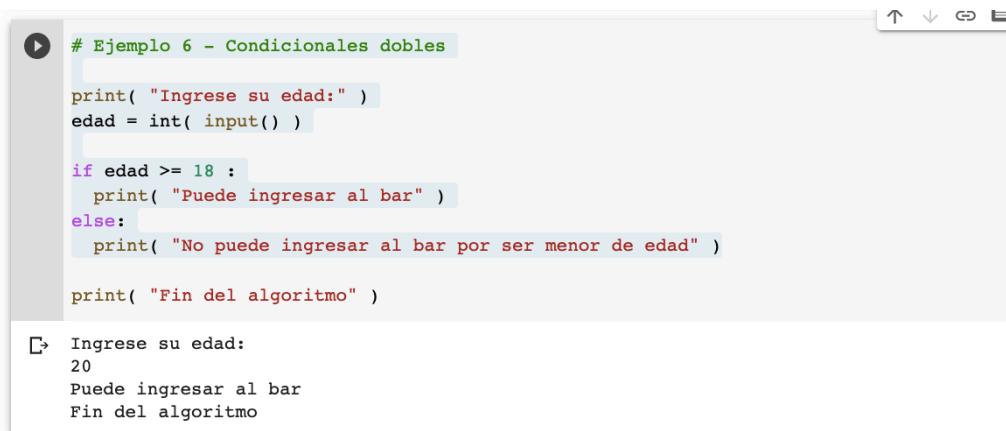
- *Datos Entrada:* edad
- *Datos Salida:* mensaje

```
# Ejemplo 6 - Condicionales dobles

print( "Ingrese su edad:" )
edad = int( input() )

if edad >= 18 :
    print( "Puede ingresar al bar" )
else:
    print( "No puede ingresar al bar" )
```

### Caso de Prueba 1: Comparación con resultado True



```
# Ejemplo 6 - Condicionales dobles

print( "Ingrese su edad:" )
edad = int( input() )

if edad >= 18 :
    print( "Puede ingresar al bar" )
else:
    print( "No puede ingresar al bar por ser menor de edad" )

print( "Fin del algoritmo" )

Ingrese su edad:
20
Puede ingresar al bar
Fin del algoritmo
```

### Caso de Prueba 2: Comparación con resultado False



```
# Ejemplo 6 - Condicionales dobles

print( "Ingrese su edad:" )
edad = int( input() )

if edad >= 18 :
    print( "Puede ingresar al bar" )
else:
    print( "No puede ingresar al bar por ser menor de edad" )

print( "Fin del algoritmo" )

Ingrese su edad:
15
No puede ingresar al bar por ser menor de edad
Fin del algoritmo
```

### Caso de Prueba 3: Comparación con resultado True (comparando valores iguales)

```
# Ejemplo 6 - Condicionales dobles

print( "Ingrese su edad: " )
edad = int( input() )

if edad >= 18 :
    print( "Puede ingresar al bar" )
else:
    print( "No puede ingresar al bar por ser menor de edad" )
|
print( "Fin del algoritmo" )

⇒ Ingrese su edad:
18
Puede ingresar al bar
Fin del algoritmo
```

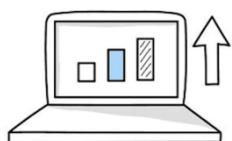
### Ejercicios Propuestos



Modifique el ejemplo 3 para que ahora despliegue un mensaje al estudiante que le indique cuando no aprueba el examen

Modifique el ejemplo 4 para que considere el siguiente caso: si el salario es igual o superior a \$1.000.000, entonces el aumento es del 2.5%

### Ejemplo 7



En un parque de diversiones, una atracción denominada el barco pirata permite su ingreso si la estatura de la persona es superior a 1.55 metros. Escribir un algoritmo que permita leer la estatura de una persona y muestre un mensaje que indique si puede ingresar o no a la atracción.

### Solución

- *Datos Entrada:* estatura
- *Datos Salida:* mensaje

```
#Solucion ejemplo 1
print( "Ingrese la estatura:" )
estatura = float ( input() )

if estatura > 1.55 :
    print( "puede ingresar")
else:
    print ( "No puede ingresar ")
```

*Caso de prueba 1. Cuando la condición es True*



```
#Solucion ejemplo 1
print( "Ingrese la estatura:" )
estatura = float ( input() )

if estatura > 1.55 :
    print( "puede ingresar")
else:
    print ( "No puede ingresar ")

▷ Ingrese la estatura:
1.56
puede ingresar
```

*Caso de prueba 2. Cuando la condición es False*

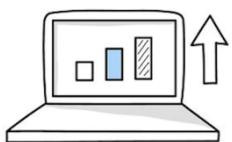


```
#Solucion ejemplo 1
print( "Ingrese la estatura:" )
estatura = float ( input() )

if estatura > 1.55 :
    print( "puede ingresar")
else:
    print ( "No puede ingresar ")

▷ Ingrese la estatura:
1.53
No puede ingresar
```

## Ejemplo 8



Escribir un programa que lea dos números y determine cuál es el mayor. Asuma que los números no serán iguales.

### Solución

- *Datos Entrada:* numero1, numero2
- *Datos Salida:* mensaje

```

print( "Ingrese el primer número: " )
numero1 = int( input( ) )
print( "Ingrese el segundo número: " )
numero2 = int( input( ) )

if numero1 > numero2 :
    print( "El mayor es: ", numero1 )
else:
    print( "El mayor es: ", numero2 )|
```

*Caso de prueba 1. Cuando la condición es True*



```

# Ejercicio Propuesto 2

# Datos de Entrada
print( "Ingrese el primer número: " )
numero1 = int( input( ) )
print( "Ingrese el segundo número: " )
numero2 = int( input( ) )

if numero1 > numero2 :
    print( "El mayor es: ", numero1 )
else:
    print( "El mayor es: ", numero2 )|
```

▷ Ingrese el primer número:  
5  
Ingrese el segundo número:  
2  
El mayor es: 5

*Caso de prueba 2. Cuando la condición es False*

```

# Ejercicio Propuesto 2

# Datos de Entrada
print( "Ingrese el primer número: " )
numero1 = int( input( ) )
print( "Ingrese el segundo número: " )
numero2 = int( input( ) )

if numero1 > numero2 :
    print( "El mayor es: ", numero1 )
else:
    print( "El mayor es: ", numero2 )

⇒ Ingrese el primer número:
3
Ingrese el segundo número:
7
El mayor es: 7

```

*Caso de prueba 3. Cuando la condición es False (Los valores a comparar son iguales)*

```

# Ejercicio Propuesto 2

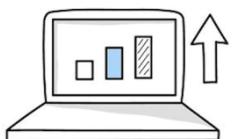
# Datos de Entrada
print( "Ingrese el primer número: " )
numero1 = int( input( ) )
print( "Ingrese el segundo número: " )
numero2 = int( input( ) )

if numero1 > numero2 :
    print( "El mayor es: ", numero1 )
else:
    print( "El mayor es: ", numero2 )

⇒ Ingrese el primer número:
5
Ingrese el segundo número:
5
El mayor es: 5

```

## Ejemplo 9



Escribir un programa que lea dos números y determine cuál es el mayor. Asuma que los números no serán iguales.

### Solución

- *Datos Entrada:* dia, precioBoleta
- *Datos Salida:* valorPagar

```

print( "Ingrese el día: " )
dia = input( )

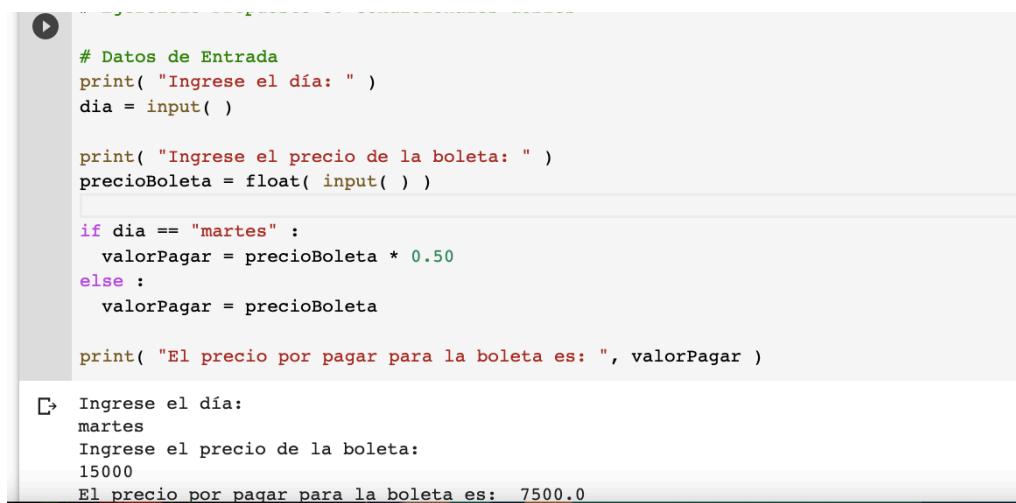
print( "Ingrese el precio de la boleta: " )
precioBoleta = float( input( ) )

if dia == "martes" :
    valorPagar = precioBoleta * 0.50
else :
    valorPagar = precioBoleta

print( "El precio por pagar para la boleta es: ", valorPagar )

```

*Caso de prueba 1. Cuando la condición es True*



```

# Datos de Entrada
print( "Ingrese el día: " )
dia = input( )

print( "Ingrese el precio de la boleta: " )
precioBoleta = float( input( ) )

if dia == "martes" :
    valorPagar = precioBoleta * 0.50
else :
    valorPagar = precioBoleta

print( "El precio por pagar para la boleta es: ", valorPagar )

```

→ Ingrese el día:  
martes  
Ingrese el precio de la boleta:  
15000  
El precio por pagar para la boleta es: 7500.0

*Caso de prueba 2. Cuando la condición es False*

```

# Datos de Entrada
print( "Ingrese el día: " )
dia = input( )

print( "Ingrese el precio de la boleta: " )
precioBoleta = float( input( ) )

if dia == "martes" :
    valorPagar = precioBoleta * 0.50
else :
    valorPagar = precioBoleta

print( "El precio por pagar para la boleta es: ", valorPagar )

▷ Ingrese el día:
jueves
Ingrese el precio de la boleta:
20000
El precio por pagar para la boleta es: 20000.0

```

*Caso de prueba 3. Cuando la condición es False (el día no es válido)*

```

# Datos de Entrada
print( "Ingrese el día: " )
dia = input( )

print( "Ingrese el precio de la boleta: " )
precioBoleta = float( input( ) )

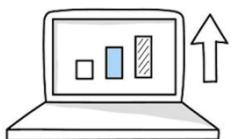
if dia == "martes" :
    valorPagar = precioBoleta * 0.50
else :
    valorPagar = precioBoleta

print( "El precio por pagar para la boleta es: ", valorPagar )

▷ Ingrese el día:
pepito
Ingrese el precio de la boleta:
10000
El precio por pagar para la boleta es: 10000.0

```

## Ejemplo 10



Determinar mediante un algoritmo si una persona puede ejercer el derecho al voto, sabiendo que lo puede hacer en caso de que sea mayor de edad.

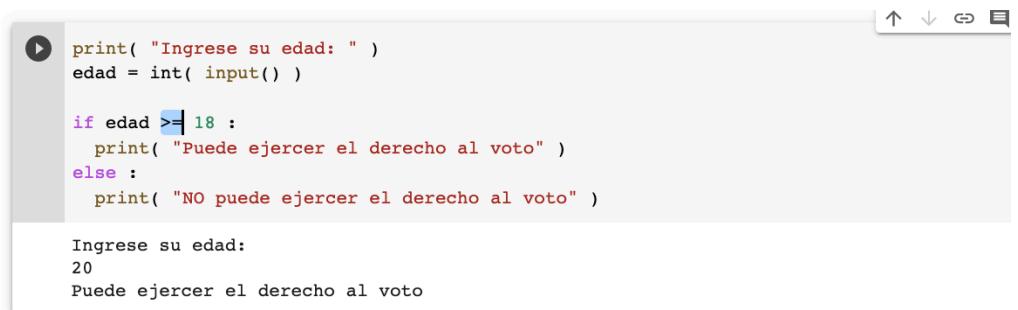
### Solución

- *Datos Entrada:* edad
- *Datos Salida:* mensaje

```
print( "Ingrese su edad: " )
edad = int( input() )

if edad >= 18 :
    print( "Puede ejercer el derecho al voto" )
else :
    print( "NO puede ejercer el derecho al voto" )
```

*Caso de prueba 1: La condición es verdadera (la edad es mayor a 18)*



```
▶ print( "Ingrese su edad: " )
edad = int( input() )

if edad >= 18 :
    print( "Puede ejercer el derecho al voto" )
else :
    print( "NO puede ejercer el derecho al voto" )

Ingrese su edad:
20
Puede ejercer el derecho al voto
```

*Caso de prueba 2: La condición es verdadera (la edad es 18)*

```

    print( "Ingrese su edad: " )
edad = int( input() )

if edad >= 18 :
    print( "Puede ejercer el derecho al voto" )
else :
    print( "NO puede ejercer el derecho al voto" )

Ingrese su edad:
18
Puede ejercer el derecho al voto

```

*Caso de prueba 3: La condición es falsa (la edad es menor a 18)*

```

    print( "Ingrese su edad: " )
edad = int( input() )

if edad >= 18 :
    print( "Puede ejercer el derecho al voto" )
else :
    print( "NO puede ejercer el derecho al voto" )

Ingrese su edad:
17
NO puede ejercer el derecho al voto

```

Ahora bien, la condición se puede expresar de otra forma:

```

print( "Ingrese su edad: " )
edad = int( input() )

if edad < 18 :
    print( "NO puede ejercer el derecho al voto" )
else :
    print( "Puede ejercer el derecho al voto" )

```

*Caso de prueba 1: La condición es verdadera (la edad es menor a 18)*

```

    print( "Ingrese su edad: " )
edad = int( input() )

if edad < 18 :
    print( "NO puede ejercer el derecho al voto" )
else :
    print( "Puede ejercer el derecho al voto" )

▷ Ingrese su edad:
17
NO puede ejercer el derecho al voto

```

*Caso de prueba 2: La condición es falsa (la edad es mayor a 18)*

```

    print( "Ingrese su edad: " )
edad = int( input() )

if edad < 18 :
    print( "NO puede ejercer el derecho al voto" )
else :
    print( "Puede ejercer el derecho al voto" )

▷ Ingrese su edad:
25
Puede ejercer el derecho al voto

```

*Caso de prueba 3: La condición es falsa (la edad es igual a 18)*

```

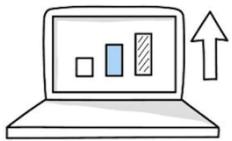
    print( "Ingrese su edad: " )
edad = int( input() )

if edad < 18 :
    print( "NO puede ejercer el derecho al voto" )
else :
    print( "Puede ejercer el derecho al voto" )

▷ Ingrese su edad:
18
Puede ejercer el derecho al voto

```

## Ejemplo 11.



Escriba un script que permita determinar si un número es par o no. Un número es par si es divisible por 2, es decir, su residuo es cero. Nota. Recuerde que en Python el operador para calcular el residuo es %. (Ejemplo. numero1 % 2, significa que está dividiendo el valor de numero1 entre 2 y tomando su residuo)

### Solución

En primer lugar, se determina que un número par es cuando el mismo se divide entre 2 y su residuo es 0. El otro residuo posible es 1, lo cual establece que se trata de un número impar.

Por ejemplo,

$$\begin{array}{ll} 8 / 2 = 4 \text{ y residuo } 0 & 8 \text{ es Par} \\ 7 / 2 = 3 \text{ y residuo } 1 & 7 \text{ es Impar} \end{array}$$

En una división existen dos resultados: el cociente y el residuo. Cada uno de estos se puede obtener con un operador diferente. Así entonces,

- |   |                               |
|---|-------------------------------|
| / | División, obtiene el cociente |
| % | División, obtiene el residuo  |

Por tanto,

$$\begin{array}{ll} \text{numero \% 2 == 0} & \text{Par} \\ \text{numero \% 2 != 0} & \text{Impar} \end{array}$$

- *Datos Entrada:* numero
- *Datos de Salida:* mensaje

El código en Python es el siguiente:

```

print('ingrese un numero ya sea par o impar:')
numero = int(input())

# proceso
if numero % 2 == 0 :
    print('El numero es par')

else :
    print('Este numero es impar')

```



## Estructuras condicionales múltiples (en cascada)



Es aquella en la cual existen más de dos casos y todos son mutuamente excluyentes entre sí, es decir, si un caso se cumple, entonces, se descartan todos los demás. Se evaluarán los casos en el orden en que se encuentran en el código, de arriba hacia abajo.

Sintaxis en Lenguaje de Algoritmo:

```

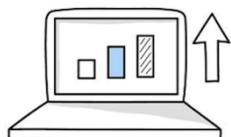
Si condición1 Entonces
    Instrucciones por ejecutar si condición1 es True
Sino Si condición2 Entonces
    Instrucciones por ejecutar si condición2 es True
        (No se cumplió condición1)
Sino Si condición3 Entonces
    Instrucciones por ejecutar si condición3 es True
        (No se cumplió condición2)
    ...
Sino
    Instrucciones por ejecutar cuando no se cumple alguna
    de las condiciones
Fin Si

```

## Sintaxis en Python:

```
if condición1 :  
    instrucciones si condición1 es True  
elif condición2 :  
    instrucciones si condición2 es True  
elif condición3 :  
    instrucciones si condición3 es True  
...  
else :  
    instrucciones si ninguna condición se cumple
```

### Ejemplo 12.



Crear un script que muestre un mensaje para cada posible color de un semáforo.

### Solución

```
# Ejemplo 1 - Instrucciones condicionales múltiples en  
cascada  
  
colorLuz = 'verde'  
  
if colorLuz == 'verde' :  
    print( "Puede pasar" )  
elif colorLuz == 'amarillo' :  
    print( "Atención, va a presentarse cambio de color" )  
else :  
    print( "Alto!!" )  
  
print( "Buen viaje" )
```

### Caso de prueba 1: Cuando el primer caso es True

```

    # Ejemplo 1 - Instrucciones condicionales múltiples en cascada

    colorLuz = 'verde'

    if colorLuz == 'verde' :
        print( "Puede pasar" )
    elif colorLuz == 'amarillo' :
        print( "Atención, va a presentarse cambio de color" )
    else :
        print( "Alto!!" )

    print( "Buen viaje" )

    ↴ Puede pasar
    Buen viaje

```

### Caso de prueba 2: Cuando el segundo caso es True

```

    # Ejemplo 1 - Instrucciones condicionales múltiples en cascada

    colorLuz = 'amarillo'

    if colorLuz == 'verde' :
        print( "Puede pasar" )
    elif colorLuz == 'amarillo' :
        print( "Atención, va a presentarse cambio de color" )
    else :
        print( "Alto!!" )

    print( "Buen viaje" )

    ↴ Atención, va a presentarse cambio de color
    Buen viaje

```

### Caso de prueba 3: Cuando el tercer caso es True

```

    # Ejemplo 1 - Instrucciones condicionales múltiples en cascada

    colorLuz = 'rojo'

    if colorLuz == 'verde' :
        print( "Puede pasar" )
    elif colorLuz == 'amarillo' :
        print( "Atención, va a presentarse cambio de color" )
    else :
        print( "Alto!!" )

    print( "Buen viaje" )

    Alto!!
    Buen viaje

```

### Caso de prueba 4: Cuando ningún caso es True

```
# Ejemplo 1 - Instrucciones condicionales múltiples en cascada

colorLuz = 'gris'

if colorLuz == 'verde' :
    print( "Puede pasar" )
elif colorLuz == 'amarillo' :
    print( "Atención, va a presentarse cambio de color" )
else :
    print( "Alto!!" )

print( "Buen viaje" )

Alto!!
Buen viaje
```

En este caso, se ha establecido un color no válido, sin embargo, dado que los dos primeros casos no se cumplen, por defecto, el flujo de ejecución ingresa por el bloque else imprimiendo el mensaje "Alto!!"

Por tanto, se modifica el código agregando un tercer caso a la estructura condicional: Cuando la luz del semáforo es de color rojo.

```
# Ejemplo 1 - Instrucciones condicionales múltiples en cascada

colorLuz = 'café'

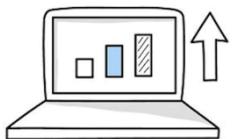
if colorLuz == 'verde' :
    print( "Puede pasar" )
elif colorLuz == 'amarillo' :
    print( "Atención, va a presentarse cambio de color" )
elif colorLuz == 'rojo':
    print( "Alto!!" )
else:
    print( "Color no válido" )

print( "Buen viaje" )

Color no válido
Buen viaje
```

Observé que se ha realizado una prueba de ejecución con un color no válido y se obtiene un mensaje más acorde.

### Ejemplo 13.



Construir un script tal que dado como dato un número entero, determine e imprima si el mismo es positivo, negativo o cero.

#### Solución

En esta situación existen 3 posibles casos:

- El número leído es positivo (es mayor a cero)
- El número leído es negativo (es menor a cero)
- El número leído es cero.

- *Datos Entrada:*      numero
- *Datos Salida:*      mensaje

El algoritmo se puede expresar de la siguiente forma:

1. Leer el número
2. Determinar si el número si es positivo, negativo o cero
3. Mostrar un mensaje indicando que tipo es

```
# Ejemplo 2 - Instrucciones condicionales múltiples
en cascada

# Datos de Entrada
print( "Ingrese el número" )
numero = int( input() )

# Proceso: Determinar que tipo de número es
if numero > 0 :
    print( "El número es positivo" )
elif numero < 0 :
    print( "El número es negativo" )
else :
    print( "El número es cero" )
```

### Caso de Prueba 1: El número es positivo

```

# Ejemplo 2 - Instrucciones condicionales múltiples en cascada

# Datos de Entrada
print( "Ingrese el número" )
numero = int( input() )

# Proceso: Determinar que tipo de número es
if numero > 0 :
    print( "El número es positivo" )
elif numero < 0 :
    print( "El número es negativo" )
else :
    print( "El número es cero" )


```

▷ Ingrese el número  
10  
El número es positivo

### Caso de Prueba 2: El número es negativo

```

# Ejemplo 2 - Instrucciones condicionales múltiples en cascada

# Datos de Entrada
print( "Ingrese el número" )
numero = int( input() )

# Proceso: Determinar que tipo de número es
if numero > 0 :
    print( "El número es positivo" )
elif numero < 0 :
    print( "El número es negativo" )
else :
    print( "El número es cero" )


```

▷ Ingrese el número  
-3  
El número es negativo

### Caso de Prueba 3: El número es cero



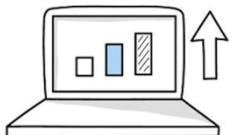
```
# Ejemplo 2 - Instrucciones condicionales múltiples en cascada

# Datos de Entrada
print( "Ingrese el número" )
numero = int( input() )

# Proceso: Determinar que tipo de número es
if numero > 0 :
    print( "El número es positivo" )
elif numero < 0 :
    print( "El número es negativo" )
else :
    print( "El número es cero" )

⇒ Ingrese el número
0
El número es cero
```

### Ejemplo 14.



Dada la calificación final de un estudiante determinar y mostrar si el mismo aprobó, reprobó o quedó en opción de habilitar. La calificación es un valor numéricico entre 0.0 y 5.0.

- Para aprobar la nota mínima es 3.0.  
nota  $\geq 3.0$
- Para reprobar, la nota debe estar por debajo de 2.5.  
nota  $< 2.5$
- Para habilitar, la nota estará entre 2.5 y 2.9.  
nota  $\geq 2.5$  y nota  $< 3.0$

### Solución

- *Datos Entrada:* nota
- *Datos Salida:* mensaje

```

# Ejemplo 3 - Instrucciones condicionales múltiples en
cascada

# Entrada
print( "Ingrese la nota final: " )
nota = float( input() )

# Proceso: Determinar si aprueba, reprueba o queda en
opción de habilitar
if nota >= 3.0 :
    print( "Aprobó el curso" )
elif nota < 2.5 :
    print( "Reprobó el curso" )
else      # la nota es mayor o igual 2.5 y es menor a 3.0
    print( "Tiene opción de habilitar" )

```

### *Caso de Prueba 1: El estudiante aprobó el curso*



```

# Ejemplo 3 - Instrucciones condicionales múltiples en cascada

# Entrada
print( "Ingrese la nota final: " )
nota = float( input() )

# Proceso: Determinar si aprueba, reprueba o queda en opción de habilitar
if nota >= 3.0 :
    print( "Aprobó el curso" )
elif nota < 2.5 :
    print( "Reprobó el curso" )
else :      # la nota es mayor o igual 2.5 y es menor a 3.0
    print( "Tiene opción de habilitar" )

⇒ Ingrese la nota final:
4.5
Aprobó el curso

```

### *Caso de Prueba 2: El estudiante reprobó el curso*

```

# Ejemplo 3 - Instrucciones condicionales múltiples en cascada

# Entrada
print( "Ingrese la nota final: " )
nota = float( input() )

# Proceso: Determinar si aprueba, reprueba o queda en opción de habilitar
if nota >= 3.0 :
    print( "Aprobó el curso" )
elif nota < 2.5 :
    print( "Reprobó el curso" )
else :      # la nota es mayor o igual 2.5 y es menor a 3.0
    print( "Tiene opción de habilitar" )

▷ Ingrese la nota final:
1.8
Reprobó el curso

```

### *Caso de Prueba 3: El estudiante puede habilitar*

```

# Ejemplo 3 - Instrucciones condicionales múltiples en cascada

# Entrada
print( "Ingrese la nota final: " )
nota = float( input() )

# Proceso: Determinar si aprueba, reprueba o queda en opción de habilitar
if nota >= 3.0 :
    print( "Aprobó el curso" )
elif nota < 2.5 :
    print( "Reprobó el curso" )
else :      # la nota es mayor o igual 2.5 y es menor a 3.0
    print( "Tiene opción de habilitar" )

▷ Ingrese la nota final:
2.7
Tiene opción de habilitar

```

## Ejemplo 15.

Leer un número entero y determinar si se encuentra en el rango (intervalo) de 1 a 10, inclusive (incluyendo los límites).

### Solución

- *Datos Entrada:* número
- *Datos Salida:* mensaje

Observe los siguientes casos:

- Por ejemplo, si el número es 0, no está en el rango porque es menor a 1
- Por ejemplo, si el número es 15, no está en el rango porque es mayor a 10
- Por ejemplo, si el número es 4, si está en el rango porque es mayor o igual a 1, Y, menor o igual a 10

La implementación en Python es la siguiente:

```
# Ejemplo 4 - Instrucciones condicionales dobles con
# operador lógico (Y)

# Entrada
print( "Ingrese el número: " )
numero = int( input() )

# Proceso: Evaluar si el número está entre 1 y 10
if numero >= 1 and numero <= 10 :
    print( "El número está dentro del intervalo [1, 10]" )
else :
    print( "El número NO está en el intervalo [1, 10]" )
```

### Caso de Prueba 1: El número está dentro del rango

```
# Ejemplo 4 - Instrucciones condicionales dobles con operador lógico (Y)

# Entrada
print( "Ingrese el número: " )
numero = int( input() )

# Proceso: Evaluar si el número está entre 1 y 10
if numero >= 1 and numero <= 10 :
    print( "El número está dentro del intervalo [1, 10]" )
else:
    print( "El número NO está en el intervalo [1, 10]" )

⇒ Ingrese el número:
6
El número está dentro del intervalo [1, 10]
```

### Caso de Prueba 2: El número no está dentro del rango (está por debajo)

```
# Ejemplo 4 - Instrucciones condicionales dobles con operador lógico (Y)

# Entrada
print( "Ingrese el número: " )
numero = int( input() )

# Proceso: Evaluar si el número está entre 1 y 10
if numero >= 1 and numero <= 10 :
    print( "El número está dentro del intervalo [1, 10]" )
else:
    print( "El número NO está en el intervalo [1, 10]" )

⇒ Ingrese el número:
-3
El número NO está en el intervalo [1, 10]
```

### Caso de Prueba 3: El número no está dentro del rango (está por encima)

```
# Ejemplo 4 - Instrucciones condicionales dobles con operador lógico (Y)

# Entrada
print( "Ingrese el número: " )
numero = int( input() )

# Proceso: Evaluar si el número está entre 1 y 10
if numero >= 1 and numero <= 10 :
    print( "El número está dentro del intervalo [1, 10]" )
else:
    print( "El número NO está en el intervalo [1, 10]" )

⇒ Ingrese el número:
14
El número NO está en el intervalo [1, 10]
```



## Ejercicios Propuestos

### Ejercicio 1

En una tienda efectúan un descuento a los clientes dependiendo del valor de la compra. El descuento se efectúa con base en el siguiente criterio:

- Si el valor es menor que \$500.000, no hay descuento.
- Si el valor está comprendido entre \$500.000 y \$1.000.000 inclusive, se otorga un 5% de descuento.
- Si el valor es mayor a \$1.000.000, se otorga un 10% de descuento

Construya un script tal que dado el valor de la compra de un cliente, determine lo que el mismo debe pagar teniendo en cuenta el descuento que aplica.

### Ejercicio 2

El costo de las llamadas telefónicas internacionales a través de dispositivo móvil (celular) depende de la zona geográfica en la que se encuentre el país destino y del número de minutos hablados. En la siguiente tabla se presenta el costo del minuto por zona.

<b>País</b>	<b>Costo por minuto</b>
Estados Unidos	\$1.000
México	\$800
Venezuela	\$500
Otro	\$1.300

Escribir un script que le permita calcular e imprimir el costo total de una llamada dependiendo del país destino y el número de minutos hablados.

### Ejercicio 3

Escriba un script que permita calcular lo que hay que pagarle a un trabajador teniendo en cuenta su sueldo y las horas extras trabajadas. Para el pago de horas extras se toma en cuenta la categoría del trabajador.

Categoría	Precio Hora Extra
1	\$10.000
2	\$12.000
3	\$15.000
4	\$20.000

Cada trabajador puede tener como máximo 30 horas extras, si tienen más sólo se les pagarán 30. A los trabajadores con categoría mayor a 4 no debemos pagarle horas extras.

Por ejemplo,

Si un trabajador tiene categoría 1, su sueldo es \$2.000.000 y trabajó 5 horas extras, el valor del sueldo por pagar es:

La hora extra se paga a \$10.000 a un trabajador categoría 1

El valor por horas extras es:  $5 * \$10.000 = \$50.000$

El sueldo por pagar es:  $\$2.000.000 + \$50.000 = \$2.050.000$

### Ejercicio 4

En un hospital se ha hecho un estudio sobre los pacientes registrados durante los últimos 10 años, con el objeto de hacer una aproximación de los costos de internación por paciente. Se obtuvo un costo promedio diario según el tipo de enfermedad que aqueja al paciente. Además, se pudo determinar que en promedio todos los pacientes con edad entre 14 y 22 años implican un costo adicional del 10%. La siguiente tabla expresa los costos diarios, según el tipo de enfermedad.

<b>Tipo de Enfermedad</b>	<b>Costo diario por paciente</b>
A	\$25.000
B	\$16.000
C	\$20.000
D	\$32.000

Construya un script que calcule e imprima el costo total que representa un paciente.

### **Ejercicio Propuesto 5**

Leer 3 números y determinar cuál es el menor. Asuma que los números son diferentes.

Si se tienen tres números a, b y c, entonces a es el menor si se cumplen dos condiciones:  $a < b$  y  $a < c$

### **Ejercicio Propuesto 6**

Leer 4 números y determinar cuál es el mayor. Asuma que los números son diferentes.

